This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:		
☐ BLACK BORDERS		
☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES		
☐ FADED TEXT OR DRAWING		
☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING		
☐ SKEWED/SLANTED IMAGES		
☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS		
☐ GRAY SCALE DOCUMENTS		
☐ LINES OR MARKS ON ORIGINAL DOCUMENT		
☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY		
□ other:		

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

WEST Search History

Hide Items Restore Clear Cancel

DATE: Monday, August 09, 2004

Hide?	<u>Set</u> Name	Query	Hit Count	
	DB=E	CPAB, JPAB, DWPI, TDBD; PLUR=NO; OP=ADJ		
	L41	129 and L40	0	
	L40	context adj (id\$1 or identifier\$1)	70	
	DB=P	PGPB, USPT; PLUR=NO; OP=ADJ		
	L39	117 same L38	6	
	L38	context adj (id\$1 or identifier\$1)	480	
	L37	117 same L36	21	
	L36	state id\$1 or state identifier\$1	873	
	DB=E	$PAB,JPAB,DWPI,TDBD;\ PLUR=NO;\ OP=ADJ$		
	L35	129 and L34	0	
	L34	context adj (id or identifier)	66	
	DB=P	GPB,USPT; PLUR=NO; OP=ADJ		
	L33	117 same L32	6	
	L32	context adj (id or identifier)	422	
	DB=E	$PAB,JPAB,DWPI,TDBD;\ PLUR=NO;\ OP=ADJ$		
	L31	L29 and L30	0	
	L30	thread adj (id\$1 or identifier\$1)	97	
	L29	firmware or nanoinstruction\$1 or nano-instruction\$1 or nanocode or nano-code or nanoprogram or nano-program or L28	15526	
	L28	microinstruction\$1 or micro-instruction\$1 or microcode or micro-code or micro-operation\$1 or microoperation\$1 or micro-ops\$1 or micro-program\$1 or microprogram\$1	11202	
	DB=P	GPB,USPT; PLUR=NO; OP=ADJ		
	L27	L17 and L22	1	
	L26	957002.ap.	4	
	L25	956875.ap.	2	
	L24	tremblay.in. and treat.xp.	5	
	L23	joy.in. and treat.xp.	0	
	DB=U	SPT; PLUR=NO; OP=ADJ		
	L22	6341347.pn.	1	
	L21	5778243.pn.	1	
	DB=PGPB,USPT; PLUR=NO; OP=ADJ			
	L20	439581.ap.	5	

L19	09439581.pn.	0
L18	L12 with L17	6
DB=U	USPT,PGPB; PLUR=NO; OP=ADJ	
L17	firmware or nanoinstruction\$1 or nano-instruction\$1 or nanocode or nano-code or nanoprogram or nano-program or L16	48207
L16	microinstruction\$1 or micro-instruction\$1 or microcode or micro-code or micro-operation\$1 or microoperation\$1 or micro-op\$1 or micro-program\$1	13725
DB=F	PGPB,USPT; PLUR=NO; OP=ADJ	
L15	L9 same L12	0
L14	L9 with L12	0
L13	L7 with L12	609
L12	thread adj (id\$1 or identifier\$1)	655
L11	L9 and L10	18
L10	(instruction\$1) with L7	77
L9	(microinstruction\$1 or micro-instruction\$1)	3561
L8	(microinstruction\$1 or micro-instruction\$1) with L7	0
L7	thread adj (id or identifier)	609
L6	L1 same thread\$1	17
L5	L2 same thread\$1	0
L4	5974525.uref.	3
L3	renam\$3 same L2	3
L2	state with L1	103
L1	segment register	860

END OF SEARCH HISTORY



Subscribe (Full Service) Register (Limited Service, Free) Login

Search: • The ACM Digital Library • The Guide

+abstract:(thread id) +abstract:(micro instruction)

i di ka

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used thread id micro instruction

Found 13 of 140,980

Relevance scale

Sort results by

relevance 🔽

Save results to a Binder

Search Tips

Open results in a new

Try an <u>Advanced Search</u>
Try this search in <u>The ACM Guide</u>

Display expanded form

window

Results 1 - 13 of 13

1 Performance of Firefly RPC

M. Schroeder, M. Burrows

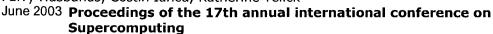
November 1989 ACM SIGOPS Operating Systems Review , Proceedings of the twelfth ACM symposium on Operating systems principles, Volume 23 Issue 5

Full text available: pdf(1.03 MB)

Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index</u> <u>terms</u>

In this paper, we report on the performance of the remote procedure call implementation for the Firefly multiprocessor and analyze the implementation to account precisely for all measured latency. From the analysis and measurements, we estimate how much faster RPC could be if certain improvements were made. The elapsed time for an inter-machine call to a remote procedure that accepts no arguments and produces no results is 2.66 milliseconds. The elapsed time for an RPC that has a ...

Compilers I: A performance analysis of the Berkeley UPC compiler Parry Husbands, Costin Iancu, Katherine Yelick



Full text available: pdf(137.75 KB)

Additional Information: full citation, abstract, references, index terms

Unified Parallel C (UPC) is a parallel language that uses a Single Program Multiple Data (SPMD) model of parallelism within a global address space. The global address space is used to simplify programming, especially on applications with irregular data structures that lead to fine-grained sharing between threads. Recent results have shown that the performance of UPC using a commercial compiler is comparable to that of MPI [7]. In this paper we describe a portable open source compiler for UPC. Ou ...

Keywords: UPC, global address space, parallel, performance

3 Implementing an efficient vector instruction set in a chip multi-processor using microthreaded pipelines

Chris Jesshope

January 2001 Australian Computer Science Communications, Proceedings of the 6th Australasian conference on Computer systems architecture, Volume 23 Issue

Full text available: pdf(911.26 KB)

Publisher Site

Additional Information: full citation, abstract, references, citings

http://portal.acm.org/results.cfm?coll=ACM&dl=ACM&CFID=25531758&CFTOKEN=54637082

This paper looks at a combination of two techniques, one of which, using a vector instruction set, has a long history dating back to pipelined vector supercomputers, such as the Cray 1 and its successors. The other technique, multi-threading, is also well understood. The novel approach proposed in this paper combines both vertical and horizontal micro-threading with vector instruction descriptors. It will be shown that a family of threads can represent a vector instruction with dependencies betw ...

4 Performance of a micro-threaded pipeline

Bing Luo, Chris Jesshope

January 2002 Australian Computer Science Communications, Proceedings of the seventh Asia-Pacific conference on Computer systems architecture -Volume 6, Volume 24 Issue 3

Full text available: 📆 pdf(650.49 KB) Additional Information: full citation, abstract, references, index terms

The micro-threaded microprocessor is a chip multi-processor, which uses a multi-threaded approach, where the threads are obtained from within a single context and exploit both vector and instruction level parallelism (ILP). This approach employs vertical and horizontal transfer in a simple pipeline. The horizontal transfer is referred to as the normal scalar pipeline processing used in most microprocessors. Vertical transfer is a context switch, which allows the code to tolerate any latency from ...

Keywords: micro thread, micro-threaded pipeline, performance evaluation, scalar pipeline

5 Multithreading and value prediction: Speculative lock elision: enabling highly concurrent multithreaded execution

Ravi Rajwar, James R. Goodman

December 2001 Proceedings of the 34th annual ACM/IEEE international symposium on Microarchitecture

Full text available: pdf(1.37 MB) Publisher Site

Additional Information: full citation, abstract, references, citings

Serialization of threads due to critical sections is a fundamental bottleneck to achieving high performance in multithreaded programs. Dynamically, such serialization may be unnecessary because these critical sections could have safely executed concurrently without locks. Current processors cannot fully exploit such parallelism because they do not have mechanisms to dynamically detect such false inter-thread dependences. We propose Speculative Lock Elision (SLE), a novel micro-architectura ...

⁶ Full-system timing-first simulation

Carl J. Mauer, Mark D. Hill, David A. Wood

June 2002 ACM SIGMETRICS Performance Evaluation Review, Proceedings of the 2002 ACM SIGMETRICS international conference on Measurement and modeling of computer systems, Volume 30 Issue 1

Full text available: pdf(87.83 KB) Additional Information: full citation, abstract, references, citings

Computer system designers often evaluate future design alternatives with detailed simulators that strive for functional fidelity (to execute relevant workloads) and performance fidelity (to rank design alternatives). Trends toward multi-threaded architectures, more complex micro-architectures, and richer workloads, make authoring detailed simulators increasingly difficult. To manage simulator complexity, this paper advocates decoupled simulator organizations that separate functiona ...

7 Slice-processors: an implementation of operation-based prediction Andreas Moshovos, Dionisios N. Pnevmatikatos, Amirali Baniasadi June 2001 Proceedings of the 15th international conference on Supercomputing Full text available: pdf(236.51 KB) Additional Information: full citation, abstract, references, citings, index terms

We describe the Slice Processor micro-architecture that implements a generalized operation-based prefetching mechanism. Operation-based prefetchers predict the series of operations, or the computation slice that can be used to calculate forthcoming memory references. This is in contrast to outcome-based predictors that exploit regularities in the (address) outcome stream. Slice processors are a generalization of existing operation-based prefetching mechanisms such as stream buffers where the ...

8 Thin locks: featherweight synchronization for Java

David F. Bacon, Ravi Konuru, Chet Murthy, Mauricio Serrano

May 1998 ACM SIGPLAN Notices, Proceedings of the ACM SIGPLAN 1998 conference on Programming language design and implementation, Volume 33 Issue 5

Full text available: pdf(1.30 MB)

Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index</u> terms

Language-supported synchronization is a source of serious performance problems in many Java programs. Even single-threaded applications may spend up to half their time performing useless synchronization due to the thread-safe nature of the Java libraries. We solve this performance problem with a new algorithm that allows lock and unlock operations to be performed with only a few machine instructions in the most common cases. Our locks only require a partial word per object, and were implemented ...

9 <u>Technical contributions: FORTH for microcomputers</u>

John S. James

October 1978 ACM SIGPLAN Notices, Volume 13 Issue 10

Full text available: pdf(749.78 KB) Additional Information: full citation, abstract, references, citings

Forth is a unique threaded language ideally suited for microcomputers. Programs are incredibly compact; e.g. in 5K to 6K bytes you can get the interactive Forth compiler, running stand-alone as its own operating system including I/O drivers and other run-time routines, plus an assembler written in Forth (in case you want to optimize time-critical programs), virtual memory software, and a text editor. Not only does all this fit into 5K to 6K bytes (of which 4K are written in Forth), but it runs i ...

10 Experimental evaluation of the Hewlett-Parkard exemplar file system

Rajesh Bordawekar, Steven Landherr, Don Capps, Mark Davis

December 1997 ACM SIGMETRICS Performance Evaluation Review, Volume 25 Issue 3

Full text available: pdf(793.88 KB) Additional Information: full citation, abstract, index terms

This article presents results from an experimental evaluation study of the HP Exemplar file system. Our experiments consist of simple micro-benchmarks that study the impact of various factors on the file system performance. These factors include I/O request/buffer sizes, vectored/non-vectored access patterns, read-ahead policies, multi-threaded (temporally irregular) requests, and architectural issues (cache parameters, NUMA behavior, etc.). Experimental results indicate that the Exemplar file s ...

11 1998: Thin locks: featherweight Synchronization for Java David F. Bacon, Ravi Konuru, Chet Murthy, Mauricio J. Serrano April 2004 ACM SIGPLAN Notices, Volume 39 Issue 4

Full text available: pdf(1.94 MB) Additional Information: full citation, abstract, references

Language-supported synchronization is a source of serious performance problems in many Java programs. Even single-threaded applications may spend up to half their time performing useless synchronization due to the thread-safe nature of the Java libraries. We solve this performance problem with a new algorithm that allows lock and unlock operations

to be performed with only a few machine instructions in the most common cases. Our locks only require a partial word per object, and were implemented ... 12 High-level scheduling model and control synthesis for a broad range of design applications Chih-Tung Chen, Kayhan Küçükçakar November 1997 Proceedings of the 1997 IEEE/ACM international conference on Computer-aided design Full text available: pdf(110.32 KB) Additional Information: full citation, abstract, references, citings, index terms Publisher Site This paper presents a versatile scheduling model and an efficient control synthesis methodology which enables architectural (high-level) design/synthesis systems to seamlessly support a broad range of architectural design applications from datapathdominated digital signal processing (DSP) to micro-processors/controllers and controldominated peripherals, utilizing multi-phase clocking schemes, multiple threading, datadependent delays, pipelining, and combinations of the above. The work present ... **Keywords**: control synthesis, scheduling model, multi-phase clocking, multi-threading, pipelining, relative scheduling, high-level synthesis, architectural synthesis, behavioral synthesis, architectural power optimization. 13 Building a high-performance communication layer over virtual interface architecture on Linux clusters Jin-Soo Kim, Kangho Kim, Sung-In Jung June 2001 Proceedings of the 15th international conference on Supercomputing Full text available: 🔁 pdf(367.79 KB) Additional Information: full citation, abstract, references, index terms The Virtual Interface Architecture (VIA) is an industry standard user-level communication architecture for cluster or system area networks. The VIA provides a protected, directlyaccessible interface to a network hardware, removing the operating system from the critical communication path. Although the VIA enables low-latency high-bandwidth communication, the application programming interface defined in the VIA specification lacks many high-level features. In this paper, we develop a ... Results 1 - 13 of 13 The ACM Portal is published by the Association for Computing Machinery. Copyright © 2004 ACM, Inc.

Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player

Real Player

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE



Membership	Publications/Services	Stan
ÜĖĖ	E Xplore	Œ

dards Conferences Careers/Jobs

Welcome

177EN	IEEE Xplore®
	1 Million Documents 1 Million Users
	zolnil Grapijace
» :	Search Results

		United States Patent and Trademark Office	1 Million Use
Help F/	AQ Terms	IEEE Peer Review Quick Links	» Search Res
Welcome	to IEEE Xplo	re*	
	hat Can Access?	Your search matched 13 of 1058483 documents. A maximum of 500 results are displayed, 15 to a page, sorted by ReDescending order.	levance in
O- Jo	Contents ournals Magazines onference oceedings	Refine This Search: You may refine your search by editing the current search expression on the text box. thread id or thread identifier <and>microinstruction or m Check to search within this result set</and>	or entering a
	andards	Results Key: JNL = Journal or Magazine CNF = Conference STD = Standard	
O- By O- Ac Member O- Jo O- Es W O- Ac	earch - By Author - Basic - Advanced ember Services - Join IEEE - Establish IEEE Web Account - Access the IEEE Member	1 A 32-bit superscalar microprocessor with 64-bit processing a bandwidth DRAM interface Matsuo, M.; Kondo, H.; Takata, Y.; Kobayashi, S.; Satoh, M.; Yoshida Y.; Hinata, JI.; Computer Design: VLSI in Computers and Processors, 1994. ICCD '94 Proceedings., IEEE International Conference on , 10-12 Oct. 1994 Pages:203 - 210 [Abstract] [PDF Full-Text (616 KB)] IEEE CNF	a, T.; Saito,
Ac O- Ac IE Fil	ccess the EE Enterpris le Cabinet	reconfigurable machine for highly parallel real-time application Erdogan, S.S.; Tunali, E.T.; Kuo, N.; Real-Time Applications, 1993 Proceedings of the IEEE Workshop on	ns
Print	Format	[Abstract] [PDF Full-Text (344 KB)] IEEE CNF	
		3 Controller synthesis for digital signal processors Owall, V.; Torkelson, M.; Circuits and Systems, 1991., IEEE International Sympoisum on , 11-1 Pages:2192 - 2195 vol.4	4 June 1991

[Abstract] [PDF Full-Text (412 KB)] IEEE CNF

4 A configurable architecture for smart pixel research

Cantin, J.F.; Beyette, F.R., Jr.;

Electronic-Enhanced Optics, Optical Sensing in Semiconductor Manufacturing, Electro-Optics in Space, Broadband Optical Networks, 2000. Digest of the LEOS Summer Topical Meetings, 24-28 July 2000

Pages: I67 - I68

[Abstract] [PDF Full-Text (256 KB)] **IEEE CNF**

5 A simulation methodology for software energy evaluation

Mehta, H.; Owens, R.M.; Irwin, M.J.;

VLSI Design, 1997. Proceedings., Tenth International Conference on , 4-7 Jan. 1997

Pages: 509 - 510

[Abstract] [PDF Full-Text (228 KB)] **IEEE CNF**

6 A 50ns floating-point signal processor VLSI

Kaneko, T.; Yamauchi, H.; Iwata, A.;

Acoustics, Speech, and Signal Processing, IEEE International Conference on

ICASSP '86., Volume: 11, Apr 1986

Pages:401 - 404

[Abstract] [PDF Full-Text (144 KB)]

7 A 55K transistor NMOS microgrammable microprocessor

Grant, B.;

Solid-State Circuits Conference, Digest of Technical Papers, 1983 IEEE

International , Volume: XXVI , Feb 1983

Pages:30 - 31

[Abstract] [PDF Full-Text (720 KB)]

8 A retargetable technique for predicting execution time

Harmon, M.G.; Baker, T.P.; Whalley, D.B.;

Real-Time Systems Symposium, 1992, 2-4 Dec. 1992

Pages: 68 - 77

[Abstract] [PDF Full-Text (716 KB)] **IEEE CNF**

9 Controller synthesis for application specific digital signal processors

Owall, V.; Torkelson, M.;

ASIC Conference and Exhibit, 1991. Proceedings., Fourth Annual IEEE

International, 23-27 Sept. 1991

Pages:P13 - 5/1-4

[Abstract] [PDF Full-Text (296 KB)] **IEEE CNF**

10 Architecture and applications of the HiPAR video signal processor

Ronner, K.; Kneip, J.;

Circuits and Systems for Video Technology, IEEE Transactions on , Volume:

6 , Issue: 1 , Feb. 1996

Pages: 56 - 66

[Abstract] [PDF Full-Text (1332 KB)]

11 APLS: active protocol label switching

Lau, W.; Jha, S.; Communications, 2003. ICC '03. IEEE International Conference on , Volume: 1, 11-15 May 2003 Pages:641 - 646 vol.1

[Abstract] [PDF Full-Text (244 KB)] **IEEE CNF**

12 A gestural instruction learning robot using information infrastructure Yamaquchi, T.; Kanazawa, N.; Akita, K.; Yoshihara, M.; Fuzzy Systems, 1995. International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium., Proceedings of 1995 IEEE International Conference on , Volume: 5 , 20-24 March 1995 Pages:87 - 90 vol.5

[PDF Full-Text (276 KB)] [Abstract] **IEEE CNF**

13 A hierarchical methodology for verifying microprogrammed microprocessors

Windley, P.J.; Research in Security and Privacy, 1990. Proceedings., 1990 IEEE Computer Society Symposium on , 7-9 May 1990 Pages: 345 - 357

[Abstract] [PDF Full-Text (968 KB)] **IEEE CNF**

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search | Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting | No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

Copyright © 2004 IEEE - All rights reserved



Subscribe (Full Service) Register (Limited Service, Free) Login

Search: • The ACM Digital Library • C The Guide

+abstract:(thread id) +abstract:(microinstruction)



Nothing Found

Your search for **+abstract:(thread id) +abstract:(microinstruction)** did not return any results.

You may want to try an Advanced Search for additional options.

Please review the Quick Tips below or for more information see the Search Tips.

Quick Tips

• Enter your search terms in lower case with a space between the terms.

sales offices

You can also enter a full question or concept in plain language.

Where are the sales offices?

 Capitalize <u>proper nouns</u> to search for specific people, places, or products.

John Colter, Netscape Navigator

• Enclose a phrase in double quotes to search for that exact phrase.

"museum of natural history" "museum of modern art"

• Narrow your searches by using a + if a search term <u>must appear</u> on a page.

museum +art

Exclude pages by using a - if a search term must not appear on a page.

museum -Paris

Combine these techniques to create a specific search query. The better your description of the information you want, the more relevant your results will be.

museum +"natural history" dinosaur -Chicago

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2004 ACM, Inc.

<u>Terms of Usage Privacy Policy Code of Ethics Contact Us</u>

Useful downloads: Adobe Acrobat Q QuickTime Windows Media Player Real Player